

## How to Use this Template

1. Make a copy [ File → Make a copy... ]
2. Rename this file: “**Capstone\_Stage1**”
3. Replace the text in green

## Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [ File → Download as PDF ]
2. Create a new GitHub repo for the capstone. Name it “**Capstone Project**”
3. Add this document to your repo. Make sure it’s named “**Capstone\_Stage1.pdf**”

---

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you’ll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

**GitHub Username:** alvaro893

# Cat me

## Description

The cutest application ever, help the world to be cuter by voting and sharing pics of cats with a nice interface application.

## Intended User

This application is for all audiences, specially cat persons :3

## Features

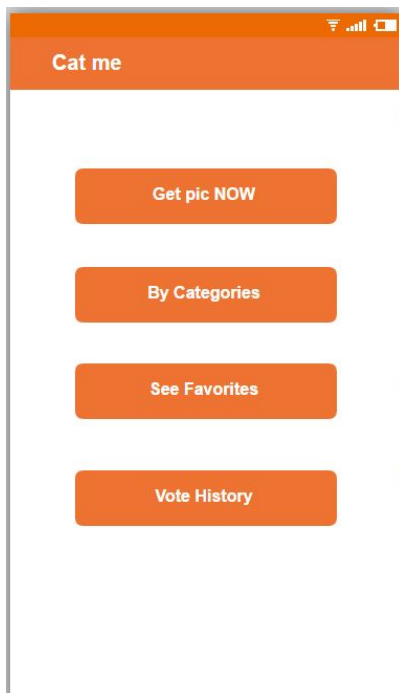
List the main features of your app. For example:

- Gets pics from this Api: <http://thecatapi.com/docs.html>
- Sends positive and negative votes
- Allows to report pics
- Saves pics to phone
- Mark a pic as your favorite
- See your voting history
- Display ads

## User Interface Mocks

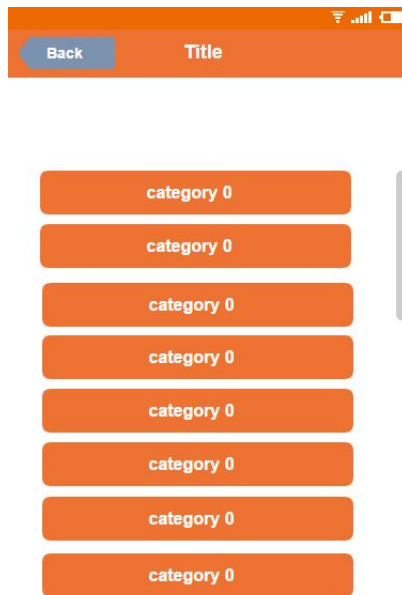
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

### Screen 1



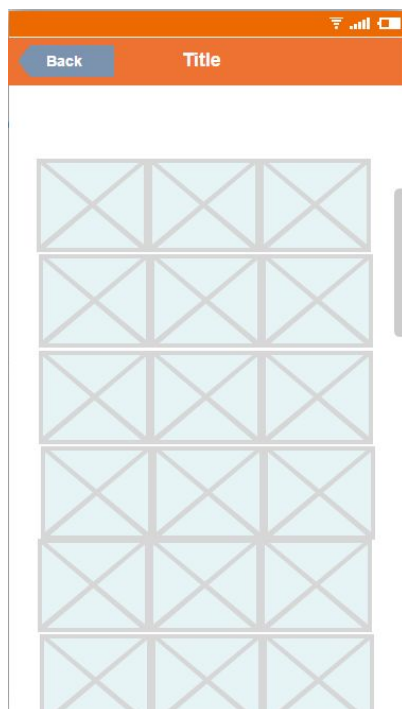
First screen menu where you can choose several options.

## Screen 2



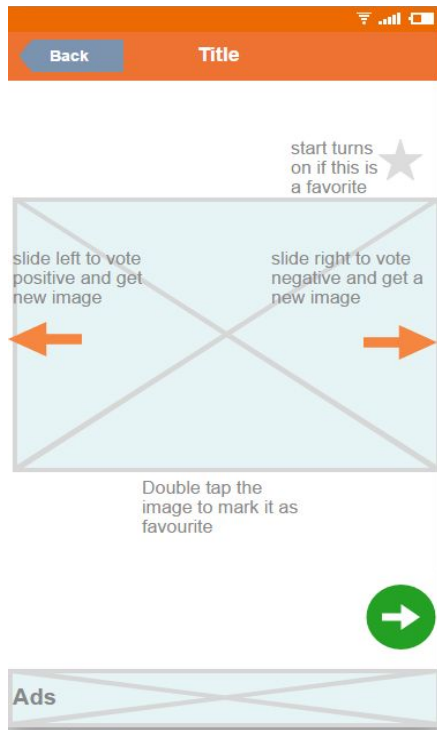
On click “By categories” on the first activity will display this. Several categories are downloaded from server, on click one will get you to the main image screen.

## Screen 3



This show a list of pictures, depending on how was accessed it will show favorites pics (on click see favorites in screen 1), or showing the vote history (on click see vote history in screen 1). When showing vote history every picture must display an icon that tell the user whether the picture was upvoted or downvoted.

## Screen 4



This is the Activity where the picture will be display. The user can upvoted sliding left or upvoting sliding right. Whatever the vote was, it will show a new random image next.

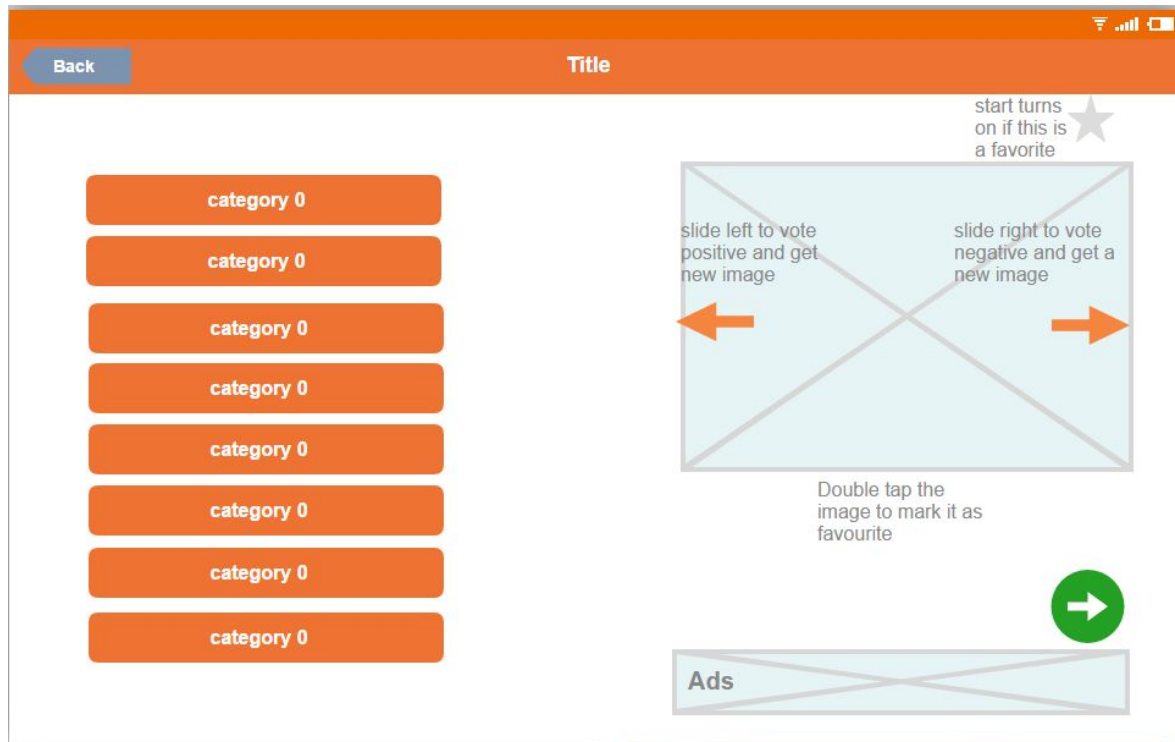
The green FAB is for share the pic.

There is an Star indicating whether the picture is favorite or not.

Double click the image will make it favorite.

All the navigation ways must reach this activity

## Screen 4



In tablet mode, when we access a list, we will see also the picture

Add as many screens as you need to portray your app's UI flow.

## Key Considerations

How will your app handle data persistence?

Database and Content provider to keep image ids, favorites and votes. Data will be send to backend and this will be sync with database to minimize the use of the network.

Describe any corner cases in the UX.

- there is no internet connection: the application still works but it does not get new images, favorited and voted images can be seen if they are still in cache.

Describe any libraries you'll be using and share your reasoning for including them.

ButterKnife, to organize better UI elements

Glide, to load the images

Retrofit, to access the Api using Http.

schematic : <https://github.com/SimonVT/schematic>. To create the content provider

**Describe how you will implement Google Play Services.**

I will use Google play service ads module, to display ads on the main screen and the analytics module as well

## Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

### Task 1: Project Setup

- Configure libraries
- Configure flavours and variants
- Create the skeleton of the application

### Task 2: Implement UI for Each Activity and Fragment

- Create UI of every activity following Material design patterns
- Make sure UI supports accessibility and localization and RLT

### Task 3: Handling data

- Create database and content provider
  - 1 Table (Picture) with fields id, server\_id, is\_favorite, and vote
- Create code to communicate with the server through http
- Handling synchronization between database and server

### Task 4: Fixing and testing

- Test application

- Fix possible bugs
- Sign apk for release

Add as many tasks as you need to complete your app.

---

### Submission Instructions

1. After you've completed all the sections, download this document as a PDF [ File → Download as PDF ]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone\_Stage1.pdf**"